Ryan Laur, Benjamin Wheeler

Lab 2 Report

Reconfigurable Computing 2

Explanation of Testbench

1. Random Test for 1000 tests
2. Consecutive Test for 200 Tests
3. Repeats 4 Times
4. Watchdog to Check for Done and Break simulation if TIMEOUT
5. Scoreboard checks internal signals against the signals used in the result model
    a. Checks if signals are cleared/reset
        i. Waits for cycle after an Active event (go = 1 when inactive), and saves the current bfm value for each signal (i_r, x_r, y_r) and puts it in the scoreboard. This is because the signals should get reset when there is an active event, but they are registered, so they will be reset on the next cycle.
    b. Checks if signal is the same upon completion
    c. Added signals into bfm from DUT.top
    d. Keeps score in scoreboard for the clearing of signals on a start event in the start monitor, and the final result of the signals using the done monitor.
6. The theoretical model signals are truncated to the OUTPUT_WIDTH before comparing with the signals from the fib entity, because an overflow check was already made.
7. Assertions
    a. Assert if go and done are both asserted, done should be cleared on the next cycle
    b. If done is asserted, but go is not asserted, done should remain true (stable)
    c. If done is cleared, then go should have been asserted on the previous cycle
    d. Go must be cleared for done to be asserted
    e. Upon completion (ie done = 1), result and overflow retain their values until the circuit is restarted

Bugs with done signal

1. Done cleared too early

Preconditions:

- DUT in done state (done == 1)
- Testbench restarted DUT via go == 1

Violation:

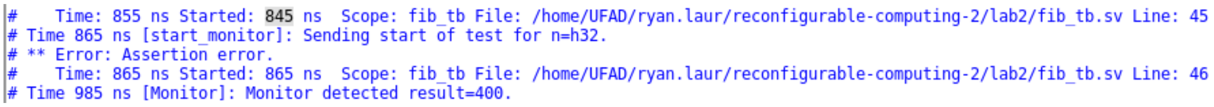
- Done not cleared on following cycle (violates design specification)

Detected in fib_tb.sv line 100:

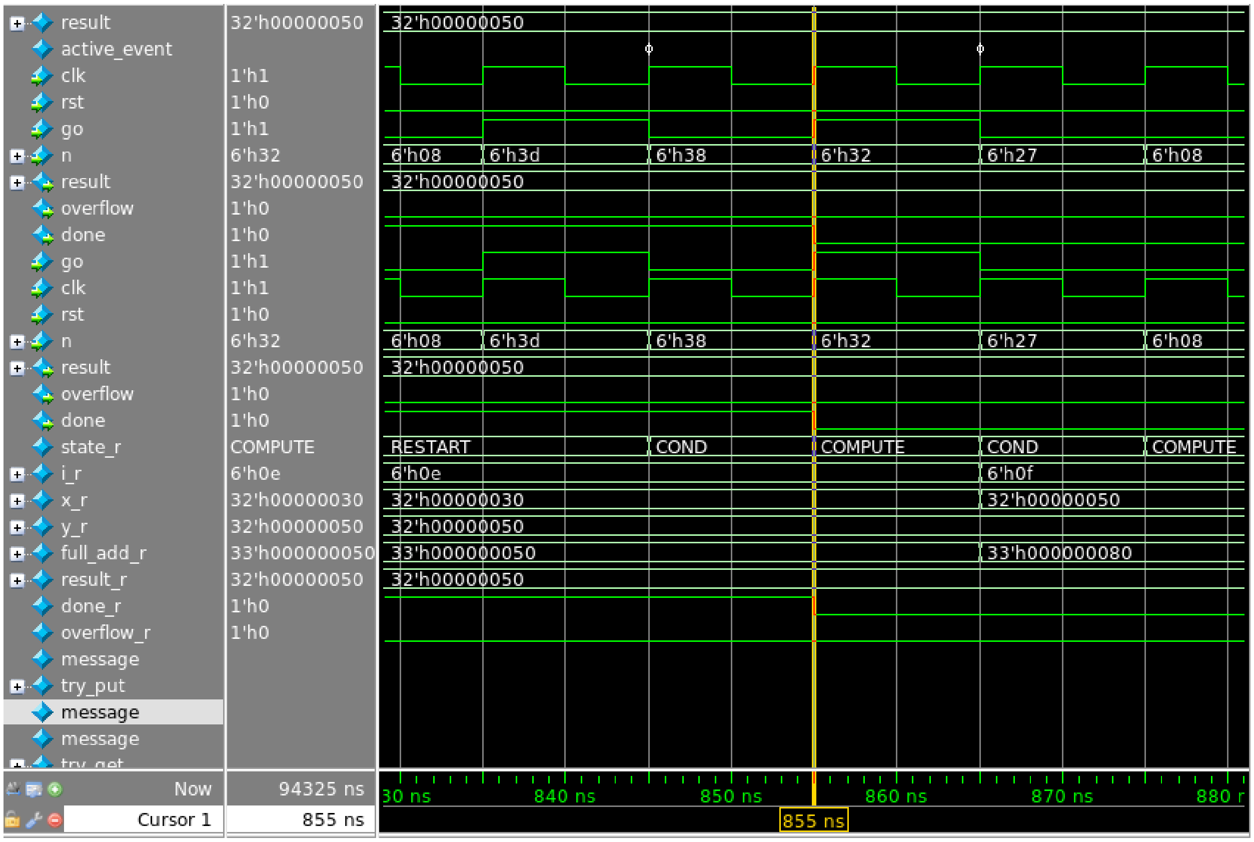
```
  // if go and done are both asserted, done should be cleared on the next cycle

assert property (@(posedge bfm.clk) disable iff (bfm.rst) bfm.go && bfm.done |=> !bfm.done)
else $error("Time %0t [Assert Property]: Done=1, go=1, done not cleared next cycle.", $time);
```

Modelsim errors:



```
#    Time: 855 ns Started: 845 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 45
# Time 865 ns [start_monitor]: Sending start of test for n=h32.
# ** Error: Assertion error.
#    Time: 865 ns Started: 865 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 46
# Time 985 ns [Monitor]: Monitor detected result=400.
```



```
result        32'h00000050
active_event
clk           1'h1
rst           1'h0
go            1'h1
n             6'h32
result        32'h00000050
overflow      1'h0
done          1'h0
go            1'h1
clk           1'h1
rst           1'h0
n             6'h32
result        32'h00000050
overflow      1'h0
done          1'h0
state_r       COMPUTE
i_r           6'h0e
x_r           32'h00000030
y_r           32'h00000050
full_add_r    33'h000000050
result_r      32'h00000050
done_r        1'h0
overflow_r    1'h0
message
try_put
message
message
try_get
Now           94325 ns
Cursor 1      855 ns
```

2. Done cleared too late

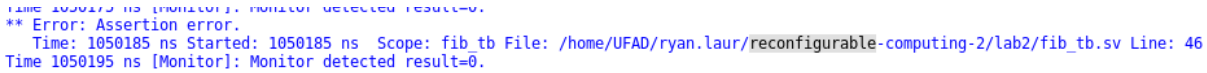
Preconditions:

- DUT in done state (done == 1)
- Testbench restarted DUT via go == 1

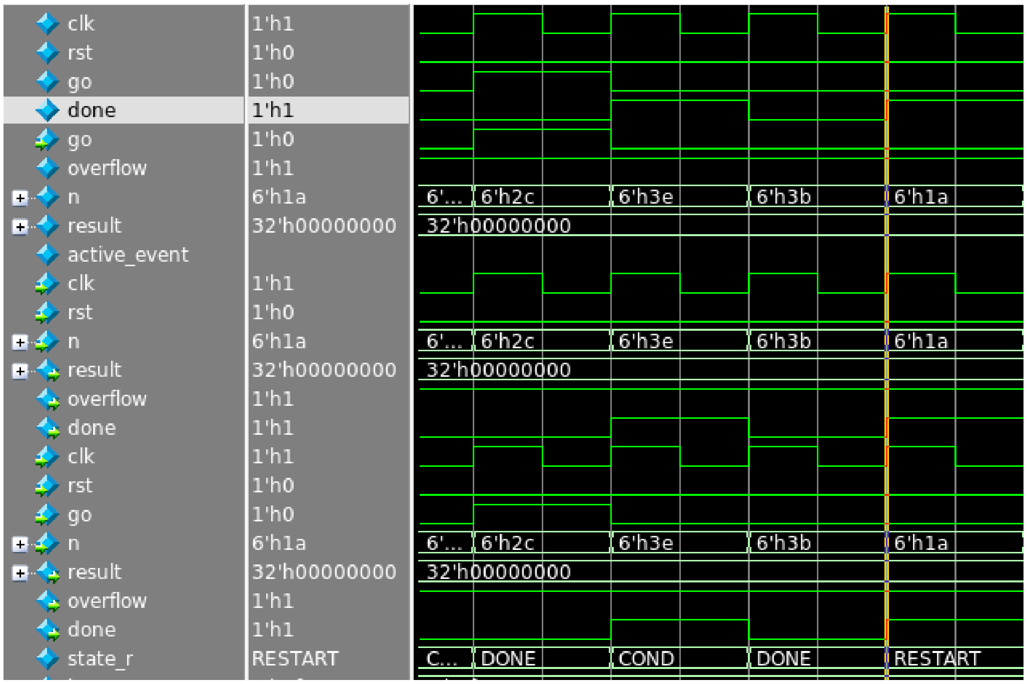
Violation:

- Done took two cycles to clear instead of one.

Detected in fib_tb.sv line 100 (see assertion property for first error).



```
Time 1050175 ns [Monitor]: Monitor detected result=0.
** Error: Assertion error.
   Time: 1050185 ns Started: 1050185 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 46
Time 1050195 ns [Monitor]: Monitor detected result=0.
```



```
clk            1'h1
rst            1'h0
go             1'h0
done           1'h1
go             1'h0
overflow       1'h1
n              6'h1a          6'... 6'h2c 6'h3e 6'h3b 6'h1a
result         32'h00000000   32'h00000000
active_event
clk            1'h1
rst            1'h0
n              6'h1a          6'... 6'h2c 6'h3e 6'h3b 6'h1a
result         32'h00000000   32'h00000000
overflow       1'h1
done           1'h1
clk            1'h1
rst            1'h0
go             1'h0
n              6'h1a          6'... 6'h2c 6'h3e 6'h3b 6'h1a
result         32'h00000000   32'h00000000
overflow       1'h1
done           1'h1
state_r        RESTART        C... DONE COND DONE RESTART
```

Infinite loop bugs

1. Max number as an input causes state machine to never finish.

Preconditions:

- Circuit fed input of max number (e.g., WIDTH’(1’b1))
    - This input is the largest number the circuit can handle for its specified width.

Violation:

- Circuit never asserts done, gets caught in infinite loop

Steps to detect the bug:

- Created a watchdog timer that reported an error and stopped the simulation if done was not asserted within 100k cycles. (monitor.svh line 131)

Reported errors from the simulation:

```
#  WATCHDOG : started at 643185
# ** Error: Time 643195 ns [Assert Property]: Done=1, go=1, done not cleared next cycle.
#    Time: 643195 ns Started: 643185 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 101
# ** Error: Time 643205 ns [Assert Property]: Done=1, go=0, done not stable.
#    Time: 643205 ns Started: 643195 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 109
# ** Error: Time 643205 ns [Assert Property]: done not cleared after go asserted.
#    Time: 643205 ns Started: 643205 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 113
#  done is not asserted time:1643180
#  WARNING::WATCHDOG BITED
# ** Note: $stop    : /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/monitor.svh(146)
#    Time: 1643180 ns  Iteration: 1  Region: /fib_tb_sv_unit::start_monitor #(6, 32)::watchdog
# Break in NamedForkStat fib_tb_sv_unit/start_monitor::watchdog/watch_dog at /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/monitor.svh line 146
```

Bug: Result register changes when circuit done

Preconditions:

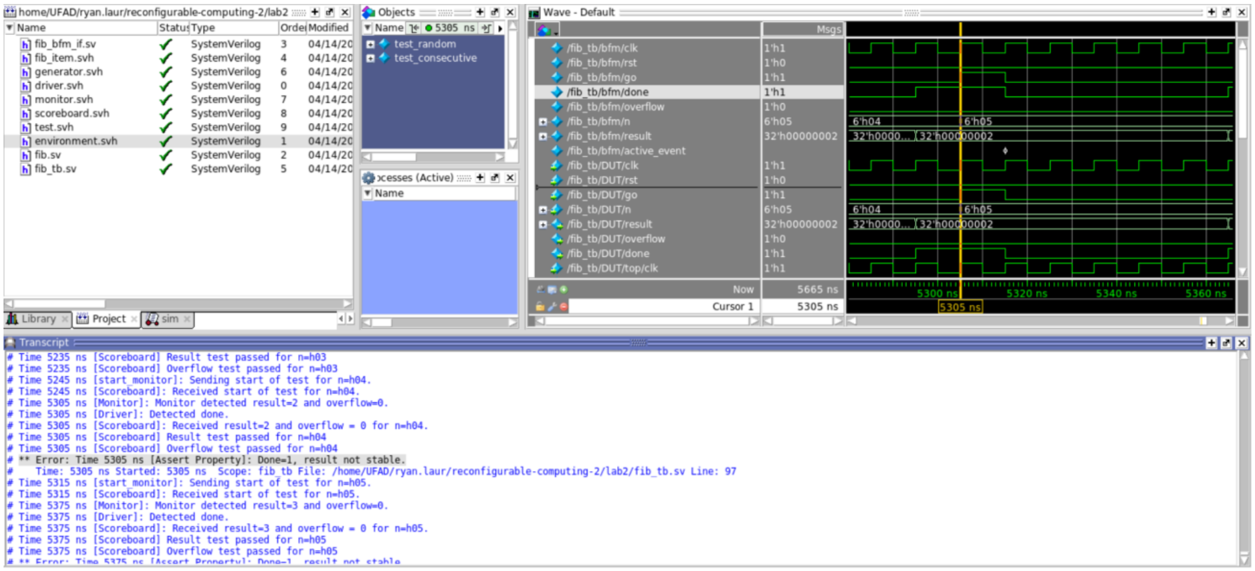
- Circuit reaches done state
- Circuit asserts done signal

Violation:

- Result register changes as done is asserted, which should not happen.

Detected in fib_tb.sv line 120, 123:

```
// upon completion, (ie done = 1), result and overflow retain their values until circuit is restarted
  assert property (@(posedge bfm.clk) disable iff (bfm.rst) bfm.done && $stable(bfm.done) |->
$stable(bfm.result))
  else $error("Time %0t [Assert Property]: Done=1, result not stable.", $time);
  // upon completion, (ie done = 1), result and overflow retain their values until circuit is restarted
  assert property (@(posedge bfm.clk) disable iff (bfm.rst) bfm.done && $stable(bfm.done) |->
$stable(bfm.overflow))
  else $error("Time %0t [Assert Property]: Done=1, overflow not stable.", $time);
```



Name	Status	Type	Order	Modified
fib_bfm_if.sv	✓	SystemVerilog	3	04/14/20
fib_item.svh	✓	SystemVerilog	4	04/14/20
generator.svh	✓	SystemVerilog	6	04/14/20
driver.svh	✓	SystemVerilog	0	04/14/20
monitor.svh	✓	SystemVerilog	7	04/14/20
scoreboard.svh	✓	SystemVerilog	8	04/14/20
test.svh	✓	SystemVerilog	9	04/14/20
environment.svh	✓	SystemVerilog	1	04/14/20
fib.sv	✓	SystemVerilog	2	04/14/20
fib_tb.sv	✓	SystemVerilog	5	04/14/20

	Msgs
/fib_tb/bfm/clk	1'h1
/fib_tb/bfm/rst	1'h0
/fib_tb/bfm/go	1'h1
/fib_tb/bfm/done	1'h1
/fib_tb/bfm/overflow	1'h0
/fib_tb/bfm/n	6'h05
/fib_tb/bfm/result	32'h00000002
/fib_tb/bfm/active_event	
/fib_tb/DUT/clk	1'h1
/fib_tb/DUT/rst	1'h0
/fib_tb/DUT/go	1'h1
/fib_tb/DUT/n	6'h05
/fib_tb/DUT/result	32'h00000002
/fib_tb/DUT/overflow	1'h0
/fib_tb/DUT/done	1'h1
/fib_tb/DUT/top/clk	1'h1
Now	5665 ns
Cursor 1	5305 ns

```
# Time 5235 ns [Scoreboard] Result test passed for n=h03
# Time 5235 ns [Scoreboard] Overflow test passed for n=h03
# Time 5245 ns [start_monitor]: Sending start of test for n=h04.
# Time 5245 ns [Scoreboard]: Received start of test for n=h04.
# Time 5305 ns [Monitor]: Monitor detected result=2 and overflow=0.
# Time 5305 ns [Driver]: Detected done.
# Time 5305 ns [Scoreboard]: Received result=2 and overflow = 0 for n=h04.
# Time 5305 ns [Scoreboard] Result test passed for n=h04
# Time 5305 ns [Scoreboard] Overflow test passed for n=h04
# ** Error: Time 5305 ns [Assert Property]: Done=1, result not stable.
#    Time: 5305 ns Started: 5305 ns  Scope: fib_tb File: /home/UFAD/ryan.laur/reconfigurable-computing-2/lab2/fib_tb.sv Line: 97
# Time 5315 ns [start_monitor]: Sending start of test for n=h05.
# Time 5315 ns [Scoreboard]: Received start of test for n=h05.
# Time 5375 ns [Monitor]: Monitor detected result=3 and overflow=0.
# Time 5375 ns [Driver]: Detected done.
# Time 5375 ns [Scoreboard]: Received result=3 and overflow = 0 for n=h05.
# Time 5375 ns [Scoreboard] Result test passed for n=h05
# Time 5375 ns [Scoreboard] Overflow test passed for n=h05
# ** Error: Time 5375 ns [Assert Property]: Done=1, result not stable.
```

Bug: Circuit reacts to changes on the input

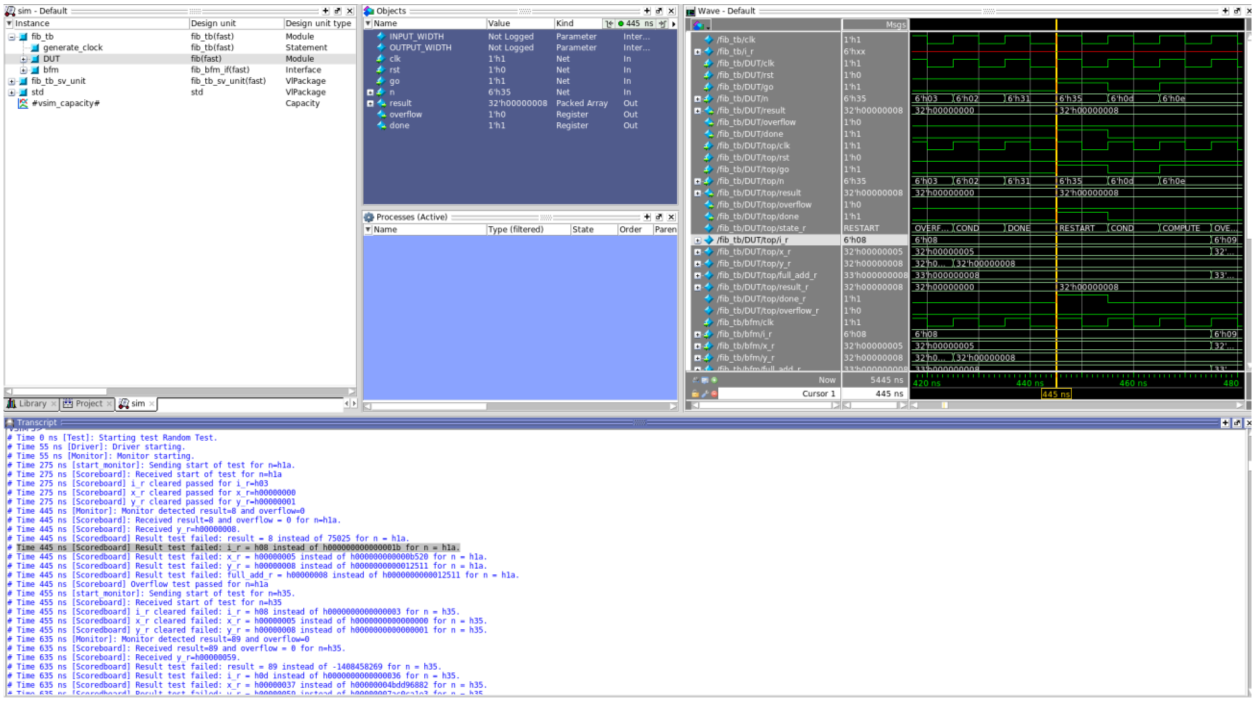
Preconditions:

- Circuit is started with go
- Input n is changed before the circuit finishes (before done == 1)

Violation:

- Circuit does not produce correct output

Discovered via scoreboard logic that ensures I, x, and y registers have correct values compared to the model function. Since they were different, this means the DUT is stopping before it should and thus producing incorrect output.



Instance	Design unit	Design unit type
fib_tb	fib_tb(fast)	Module
generate_clock	fib_tb(fast)	Statement
DUT	fib(fast)	Module
bfm	fib_bfm_if(fast)	Interface
fib_tb_sv_unit	fib_tb_sv_unit(fast)	VlPackage
std	std	VlPackage
#vsim_capacity#		Capacity

Name	Value	Kind	445 ns
INPUT_WIDTH	Not Logged	Parameter	Inter...
OUTPUT_WIDTH	Not Logged	Parameter	Inter...
clk	1'h1	Net	In
rst	1'h0	Net	In
go	1'h1	Net	In
n	6'h35	Net	In
result	32'h00000008	Packed Array	Out
overflow	1'h0	Register	Out
done	1'h1	Register	Out

	Msgs
/fib_tb/clk	1'h1
/fib_tb/i_r	6'hxx
/fib_tb/DUT/clk	1'h1
/fib_tb/DUT/rst	1'h0
/fib_tb/DUT/go	1'h1
/fib_tb/DUT/n	6'h35
/fib_tb/DUT/result	32'h00000008
/fib_tb/DUT/overflow	1'h0
/fib_tb/DUT/done	1'h1
/fib_tb/DUT/top/clk	1'h1
/fib_tb/DUT/top/rst	1'h0
/fib_tb/DUT/top/go	1'h1
/fib_tb/DUT/top/n	6'h35
/fib_tb/DUT/top/result	32'h00000008
/fib_tb/DUT/top/overflow	1'h0
/fib_tb/DUT/top/done	1'h1
/fib_tb/DUT/top/state_r	RESTART
/fib_tb/DUT/top/i_r	6'h08
/fib_tb/DUT/top/x_r	32'h00000005
/fib_tb/DUT/top/y_r	32'h00000008
/fib_tb/DUT/top/full_add_r	33'h000000008
/fib_tb/DUT/top/result_r	32'h00000008
/fib_tb/DUT/top/done_r	1'h1
/fib_tb/DUT/top/overflow_r	1'h0
/fib_tb/bfm/clk	1'h1
/fib_tb/bfm/i_r	6'h08
/fib_tb/bfm/x_r	32'h00000005
/fib_tb/bfm/y_r	32'h00000008
/fib_tb/bfm/full_add_r	33'h000000008
Now	5445 ns
Cursor 1	445 ns

```
# Time 0 ns [Test]: Starting test Random Test.
# Time 55 ns [Driver]: Driver starting.
# Time 55 ns [Monitor]: Monitor starting.
# Time 275 ns [start_monitor]: Sending start of test for n=h1a.
# Time 275 ns [Scoreboard]: Received start of test for n=h1a
# Time 275 ns [Scoreboard] i_r cleared passed for i_r=h03
# Time 275 ns [Scoreboard] x_r cleared passed for x_r=h00000000
# Time 275 ns [Scoreboard] y_r cleared passed for y_r=h00000001
# Time 445 ns [Monitor]: Monitor detected result=8 and overflow=0
# Time 445 ns [Scoreboard]: Received result=8 and overflow = 0 for n=h1a.
# Time 445 ns [Scoreboard]: Received y_r=h00000008.
# Time 445 ns [Scoredboard] Result test failed: result = 8 instead of 75025 for n = h1a.
# Time 445 ns [Scoredboard] Result test failed: i_r = h08 instead of h000000000000001b for n = h1a.
# Time 445 ns [Scoredboard] Result test failed: x_r = h00000005 instead of h000000000000b520 for n = h1a.
# Time 445 ns [Scoredboard] Result test failed: y_r = h00000008 instead of h0000000000012511 for n = h1a.
# Time 445 ns [Scoredboard] Result test failed: full_add_r = h00000008 instead of h0000000000012511 for n = h1a.
# Time 445 ns [Scoreboard] Overflow test passed for n=h1a
# Time 455 ns [start_monitor]: Sending start of test for n=h35.
# Time 455 ns [Scoreboard]: Received start of test for n=h35
# Time 455 ns [Scoredboard] i_r cleared failed: i_r = h08 instead of h0000000000000003 for n = h35.
# Time 455 ns [Scoredboard] x_r cleared failed: x_r = h00000005 instead of h0000000000000000 for n = h35.
# Time 455 ns [Scoredboard] y_r cleared failed: y_r = h00000008 instead of h0000000000000001 for n = h35.
# Time 635 ns [Monitor]: Monitor detected result=89 and overflow=0
# Time 635 ns [Scoreboard]: Received result=89 and overflow = 0 for n=h35.
# Time 635 ns [Scoreboard]: Received y_r=h00000059.
# Time 635 ns [Scoredboard] Result test failed: result = 89 instead of -1408458269 for n = h35.
# Time 635 ns [Scoredboard] Result test failed: i_r = h0d instead of h0000000000000036 for n = h35.
# Time 635 ns [Scoredboard] Result test failed: x_r = h00000037 instead of h00000004bdd96882 for n = h35.
# Time 635 ns [Scoredboard] Result test failed: y_r = h00000059 instead of h00000007ac0ca1e3 for n = h35.
```

Overflow bugs

Preconditions:

- DUT was passed an input that would cause it to overflow

Violation:

- Overflow flag not asserted

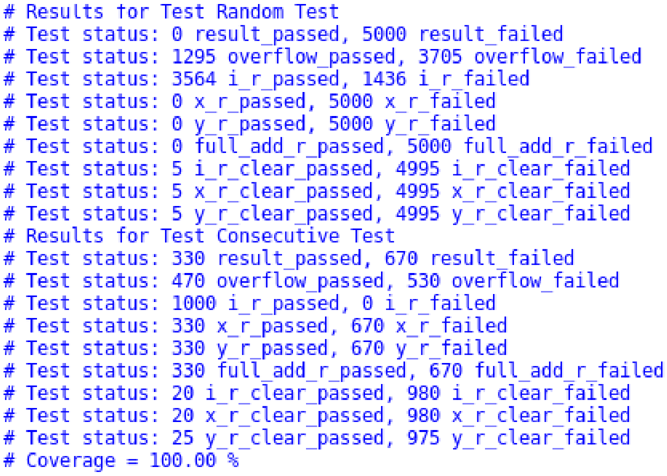
Detected by utilizing an overflow model function, which truncates the actual value of the correct result to OUTPUT_WIDTH, and compares with the original longint result. If they're different, overflow occurred and the testbench will assert that the DUT asserts its overflow flag.

Accompanying testbench error:

# Time 445 ns [Scoreboard] Result test failed: full_add_r = h00000008 instead of h1e8d0a40 for n = h31.

# Time 445 ns [Scoreboard] Overflow test failed: overflow = 0 instead of 1 for n = h31 and theoretical result = h000000011e8d0a40 and result = h00000008.

Number of Errors from Scoreboard



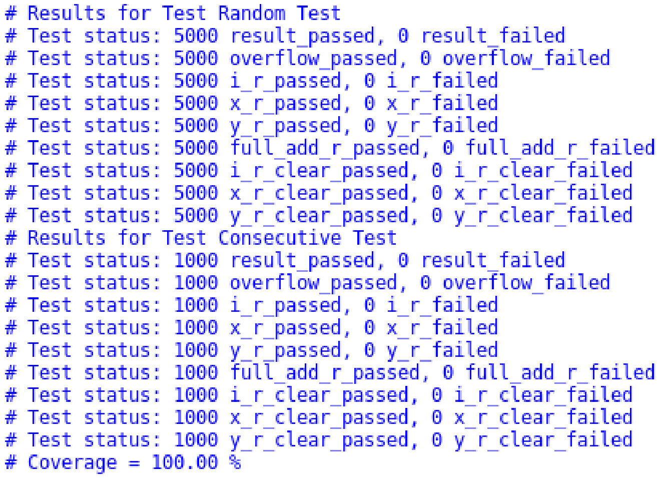
```
# Results for Test Random Test
# Test status: 0 result_passed, 5000 result_failed
# Test status: 1295 overflow_passed, 3705 overflow_failed
# Test status: 3564 i_r_passed, 1436 i_r_failed
# Test status: 0 x_r_passed, 5000 x_r_failed
# Test status: 0 y_r_passed, 5000 y_r_failed
# Test status: 0 full_add_r_passed, 5000 full_add_r_failed
# Test status: 5 i_r_clear_passed, 4995 i_r_clear_failed
# Test status: 5 x_r_clear_passed, 4995 x_r_clear_failed
# Test status: 5 y_r_clear_passed, 4995 y_r_clear_failed
# Results for Test Consecutive Test
# Test status: 330 result_passed, 670 result_failed
# Test status: 470 overflow_passed, 530 overflow_failed
# Test status: 1000 i_r_passed, 0 i_r_failed
# Test status: 330 x_r_passed, 670 x_r_failed
# Test status: 330 y_r_passed, 670 y_r_failed
# Test status: 330 full_add_r_passed, 670 full_add_r_failed
# Test status: 20 i_r_clear_passed, 980 i_r_clear_failed
# Test status: 20 x_r_clear_passed, 980 x_r_clear_failed
# Test status: 25 y_r_clear_passed, 975 y_r_clear_failed
# Coverage = 100.00 %
```

Bugs fixed in fib_good

This list includes some of the major bugs we fixed in fib_good, but is not exhaustive.

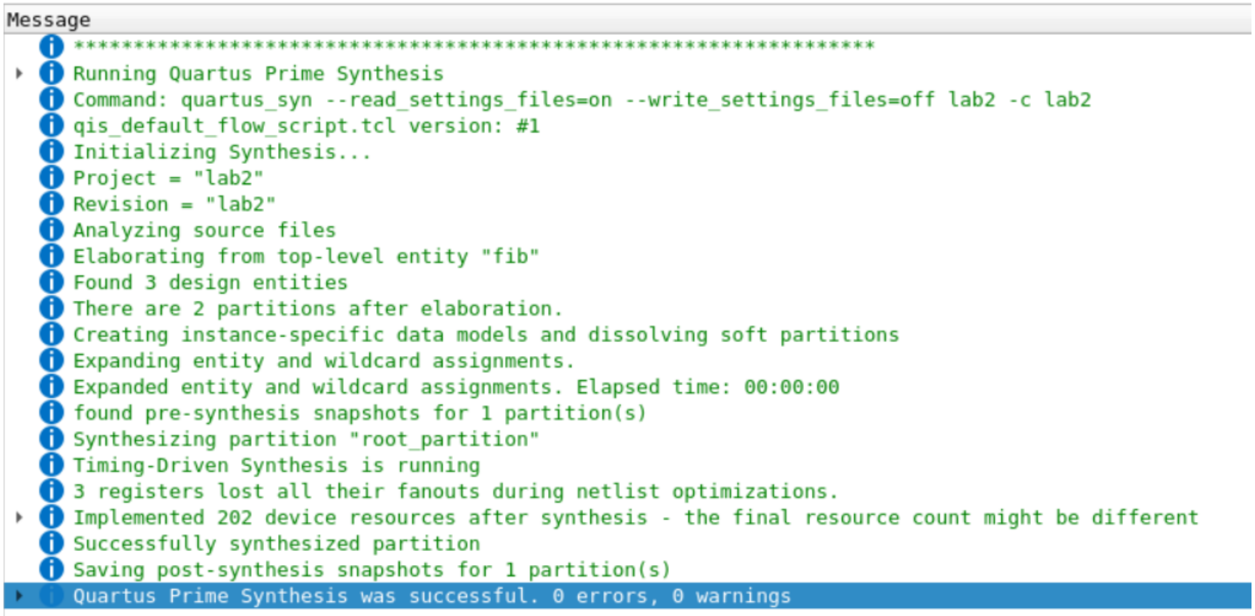
- Save circuit inputs in registers
- Reset temporary variables to the correct value on circuit reset
- Fix issues where non-blocking assignments to temp values were being read on the same clock edge
  - replaced with blocking assignments or rearranged states to ensure values were updated and read properly
- Defined state enum with logic [2:0] specifier to ensure synthesis tools recognize it as a state machine and operate/optimize accordingly
- Moved done assertions to proper states to follow guidelines on done signal
- Reset overflow on circuit reset (if overflow retains value = 1, when in fact it was a 0.)
- Removed default COND state if go = 1 outside of the case statement
- Casted to proper number of bits
- Defined that go must return to 0 to start another execution
- Define that i_r must be greater than or equal to 3 to initiate computation (solves infinite loop for i_r incrementing passed maximum number back to 0)
- Saved n as a register to prevent changes in circuit (n_r)
- Added slice for y_r <= full_add_r[OUTPUT_WIDTH-1:0] so it only takes OUTPUT_WIDTH bits

Modelsim Transcript screenshot of fib_good



```
# Results for Test Random Test
# Test status: 5000 result_passed, 0 result_failed
# Test status: 5000 overflow_passed, 0 overflow_failed
# Test status: 5000 i_r_passed, 0 i_r_failed
# Test status: 5000 x_r_passed, 0 x_r_failed
# Test status: 5000 y_r_passed, 0 y_r_failed
# Test status: 5000 full_add_r_passed, 0 full_add_r_failed
# Test status: 5000 i_r_clear_passed, 0 i_r_clear_failed
# Test status: 5000 x_r_clear_passed, 0 x_r_clear_failed
# Test status: 5000 y_r_clear_passed, 0 y_r_clear_failed
# Results for Test Consecutive Test
# Test status: 1000 result_passed, 0 result_failed
# Test status: 1000 overflow_passed, 0 overflow_failed
# Test status: 1000 i_r_passed, 0 i_r_failed
# Test status: 1000 x_r_passed, 0 x_r_failed
# Test status: 1000 y_r_passed, 0 y_r_failed
# Test status: 1000 full_add_r_passed, 0 full_add_r_failed
# Test status: 1000 i_r_clear_passed, 0 i_r_clear_failed
# Test status: 1000 x_r_clear_passed, 0 x_r_clear_failed
# Test status: 1000 y_r_clear_passed, 0 y_r_clear_failed
# Coverage = 100.00 %
```

Quartus synthesis screenshot of fib_good



```
Message
*******************************************************************
Running Quartus Prime Synthesis
Command: quartus_syn --read_settings_files=on --write_settings_files=off lab2 -c lab2
qis_default_flow_script.tcl version: #1
Initializing Synthesis...
Project = "lab2"
Revision = "lab2"
Analyzing source files
Elaborating from top-level entity "fib"
Found 3 design entities
There are 2 partitions after elaboration.
Creating instance-specific data models and dissolving soft partitions
Expanding entity and wildcard assignments.
Expanded entity and wildcard assignments. Elapsed time: 00:00:00
found pre-synthesis snapshots for 1 partition(s)
Synthesizing partition "root_partition"
Timing-Driven Synthesis is running
3 registers lost all their fanouts during netlist optimizations.
Implemented 202 device resources after synthesis - the final resource count might be different
Successfully synthesized partition
Saving post-synthesis snapshots for 1 partition(s)
Quartus Prime Synthesis was successful. 0 errors, 0 warnings
```